APSC 143 – Introduction to Programming for Engineers

Assignment 6 – Strings

Figure 1: Various Acronyms at Queens University

Objective:

Students will display their knowledge of strings and their related functions (and previous course topics) in programming using a real-world example.

Problem:

An acronym is a word formed from the initial letters or groups of letters of words in a set phrase or series of words and pronounced as a separate word. As you have probably noticed, almost everything at Queens University is an acronym. Sometimes, it is hard to keep track of all the acronyms around you. Therefore, you are going to create a program to create a database of acronyms around Queens.

Strings will be entered into the program to be turned into acronyms. Once, all the acronyms are entered, the program will accept an input of one of these acronyms, and show the full string used to create it. To create acronyms from strings for this assignment, follow the criteria below.

- Only Capital Letters are included (no lower-case letters or special characters)
- Every capital Letter must be after a space, except for if the capital letter is the first character of the string.

Example: “Faculty of Engineering & Applied Science” is“FEAS”

ASCII Theory:

ASCII is essentially a table of numerical codes that correspond to each character found on a keyboard. For example, the asterisk ***** has been assigned the decimal value 42 in ASCII. This assignment uses capital letters

from 'A' to 'Z', which are found at sequential ASCII values between 65 and 90. Note that capital and lower-case letters have different ASCII codes ('a' to 'z' have ASCII values between 91 and 116).

While it is useful to know the ASCII values for any given character, **you do not need to use them in place of regular characters** for this assignment. Remember that char type variables can be used in numerical expressions! Therefore, if you are looking for a letter between 'C' and 'K', the following sample expression will work without needing to lookup the ASCII values yourself:

```
if(letter >= 'C' && letter <= 'K') { }
```

Instructions:

Below is a guideline for how to approach this problem, though you may choose to approach it however you wish. **Make sure to express your results using the format specified below**:

- Ask User how many acronyms they want to create.
- Create **two** empty arrays to maintain the data from the program, based on the input given.
    - Array for strings to be inputted by the user.
    - Array for acronyms to be created by the program.
    - **For this assignment, you can assume the maximum length of all strings is 50.**
- Ask user to enter a string and collect it.
    - Make sure to know the difference between scanf() and fgets() to do this step correctly.
- Traverse through each character in the string and add characters to the acronym **using the criteria above.** Print the acronym to the console.
- Create a loop to ask user to enter one of the acronyms to see what it stands for.
- Use the data saved in your two arrays to see if there is a matching acronym and print the full string to the console.
- Loop back to ask the user for another acronym.
    - **If the user enters 0, the program ends.**

Comments are mandatory for this assignment. Add comments as necessary for key pieces in your code, such as variable declaration, conditional statements, and looping conditions to explain what the program is doing.

Your output must match the sample output below exactly; otherwise, the auto grading software will not be able to grade your assignment, which may affect your mark.

Example Outputs:

(Note: You **DO NOT** need to print the values in bold; they are shown only to display the **fgets** input for this example.)

(Note: All the words in the input string are on the same line.)

(Note: Make sure that and that the last line also prints a new line for Gradescope to give full credit.)

```
Enter number of acronyms to add to the database: 2
```

APSC 143 – Introduction to Programming for Engineers

```
Enter the string to convert into an acronym: Student On Line University Service

Acronym of "Student On Line University Service" is "SOLUS".

Enter the string to convert into an acronym: the Athletic & Recreation Centre

Acronym of "the Athletic & Recreation Centre" is "ARC".

Enter the acronym of the string you would like to see, or enter 0 to exit:

SOLUS

The corresponding string to "SOLUS" is "Student On Line University Service".

Enter the acronym of the string you would like to see, or enter 0 to exit: 0
```

Submission Instructions:

Create your program using CLion and upload it to Gradescope for grading. **Your program file must be named** "**apsc143assign6.c**" in order for your assignment to be graded. Do not include any personal information (student number, name, etc.) in your submission. Also, please include a comment in your code attesting to the originality of your work.

Refer to the assignment rubric on OnQ for a detailed breakdown of the grading criteria. Your submission must adhere to the assignment rules as outlined in the submission policy document for this course, which can also be found on OnQ. There is zero tolerance for plagiarism in this course. This auto grading software will automatically flag potential cases of plagiarism, which will be reviewed by the instructors.

More information on assignment submissions can be Found in Week 2, and information on the specific definition and repercussions of plagiarism can be found in the "Begin Here (About This Course)" module